# HOME CREDIT DEFAULT RISK

Vinayak Chaturvedi
MT2020046
Computer Science & Engineering
International Insitiute of Information Technology, Bangalore
Bangalore, India

Rushikesh Jachak
MT2020126
Computer Science & Engineering
International Insitiute of Information Technology, Bangalore
Bangalore, India

Swapnil Jain
MT2020171
Computer Science & Engineering
International Insitiute of Information Technology, Bangalore
Bangalore, India

***Abstract*—This is a detailed report on our work on classification to ensure that clients capable of repayment are not rejected and that loans are given with a principal, maturity, and repayment calendar that will empower their clients to be successful.**

***Keywords—Machine Learning, Data Preprocessing, Exploratory Data Analysis, Feature Engineering, Memory Optimization, Random Forest, LightGBM Boosting, Blending.***

## I. Introduction

Many people struggle to get loans due to insufficient or non-existent credit histories. And, unfortunately, this population is often taken advantage of by untrustworthy lenders.

Home Credit strives to broaden financial inclusion for the unbanked population by providing a positive and safe borrowing experience. In order to make sure this underserved population has a positive loan experience, Home Credit makes use of a variety of alternative data--including telco and transactional information--to predict their clients' repayment abilities.

## II. DATASET

This dataset contains 7 tables viz.

- application{train/test}.csv: This is the main table, broken into two files for Train (with TARGET) and Test (without TARGET).
- bureau.csv: All client's previous credits provided by other financial institutions was in this table.
- bureau_balance.csv: Monthly balances of previous credits in Credit Bureau.
- POS_CASH_balance.csv: monthly balance snapshots of previous POS (point of sales) and cash loans that the applicant had with Home Credit.
- credit_card_balance.csv: Monthly balance snapshots of previous credit cards that the applicant has.
- previous_application.csv: All previous applications for Home Credit loans of clients who have loans in our sample.
- installments_payments.csv: Repayment history for the previously disbursed credits in Home Credit related to the loans in our sample.

The application_train dataset contains 199,882 samples while test dataset contains 107,629 samples. The dataset contains 122 features among which 17 are categorial while 105 are numerical. The train dataset also contains an additional column ‘target which is the target variable.

## III. EXPLORATORY DATA ANALYSIS (*HEADING 1*)

Each row in the dataset is uniquely identified by “SKID_CURR” and whether it is able to pay loan or not is represented by the given target label. The target label 0 indicates it is able to pay loan while target label 1 indicates it is not been able to pay loan.

Percentage of positive target value 1 is 8.1203 % and Percentage of negative target value 0 is 91.8797 % as show

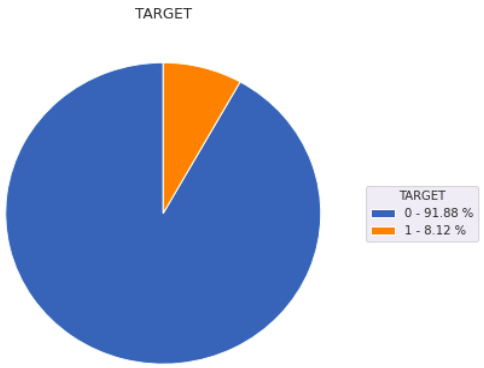

Fig. 1. Distribution of Target Label

XXX-X-XXXX-XXXX-X/XX/$XX.00 ©20XX IEEE

The data distribution of target label is skewed/ unbalanced as 92% of the target label is 0 while 8% of it is 1 as shown in Fig 1. This is kind of analogous to increasing risk measures taken by analyst team in recent years. Some of the columns with high percentage of Null values in application_train dataset are shown in Fig 2.

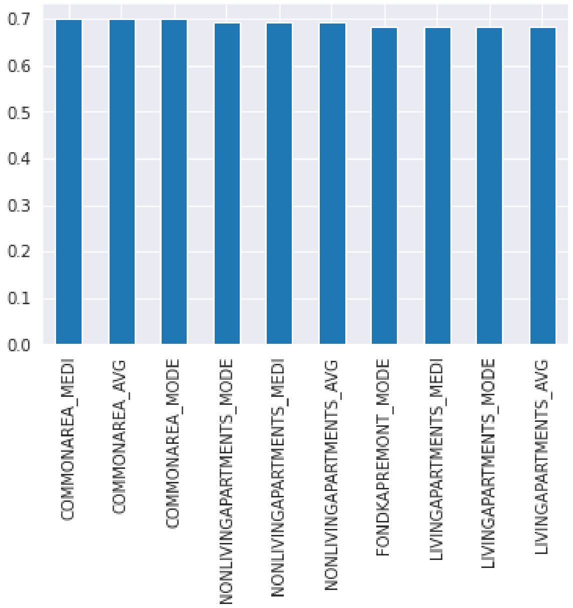

Fig. 2. Null Value Distribution of Application Train

Previous Applications received by home were fairly distrusted among all days of the week except for Thursday (lesser than ~5% compared to other days) as shown in Fig 3.

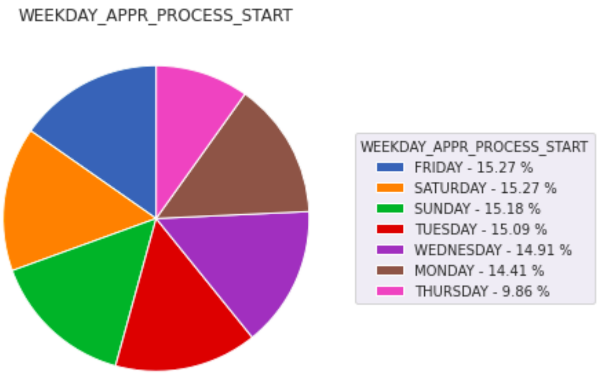

Fig. 3. Start day of Applcication

Previous application had high demand of Consumer and Cash loans as compared to Revolving loans as shown in Fig 4.

- Consumer loans: 44.76%
- Cash loans: 43.66%
- Revolving loans: 11.57 =%

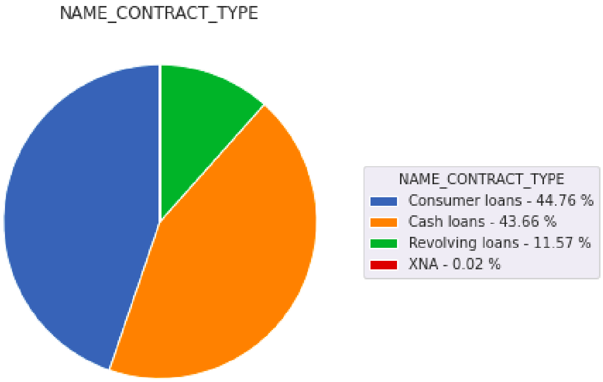

Fig. 4. Type of Contract

More than half of the previous application got approved as shown in Fig 5..

- Approved: 62.07%
- Refused + Canceled: 36.34%
- Unused offer: 1.58%.

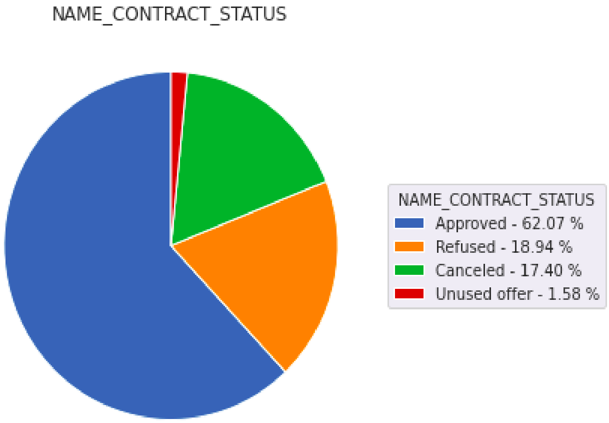

Fig. 5. Status of Contract

Most of the individuals who are taking loan are the one who has their highest education as Secondary Education and a fair amount of them are unable to repay loan as shown in Fig 6.

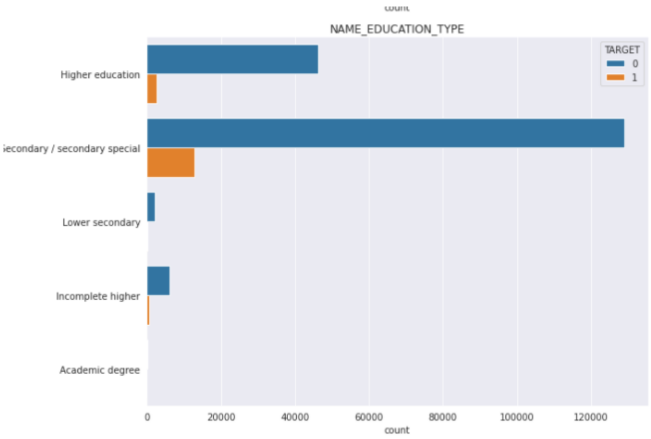

Fig. 6. Education Type of Applicantts

## IV. Data PreProcessing

### A. *Handling Missing and Outlier Values*

1. Columns with more than 60% of null values are dropped, since the dataset is highly unbalanced with a ratio of 85:15 and hence, this columns fail to contribute to target prediction.
2. The Null Values in Numerical Columns is replaced by Median of the train dataset.
3. The Null Values in Categorical Columns is replaced by Mode of the test dataset.
4. Numerical: The Numeric Columns having values outside the Inter-Quantile range are filled with NA values and then replaced by median of the dataset

### B. *Data Encoding*

1. One Hot Encoding: The Categorical Values having less than 100 unique values are encoded using One Hot Encoding Technique.
2. Frequency Encoding: Although Label Encoding is the common choice for the features that are having high cardinality, but it often gives a certain order which might not be preferable. Moreover, it is expected that individuals from same set of background are expected to perform similar on credit risk. Therefore, Features such as **Organization Type** and **Occupation Type** are frequency encoded.

### C. *Correlation Analysis*

The features which are highly corelated either positive or negative doesn't contribute much to target prediction and are being dropped. The Threshold is kept at 0.9. Any feature which is related with other with a threshold greater than 0.9 or less than 0.9 is dropped.

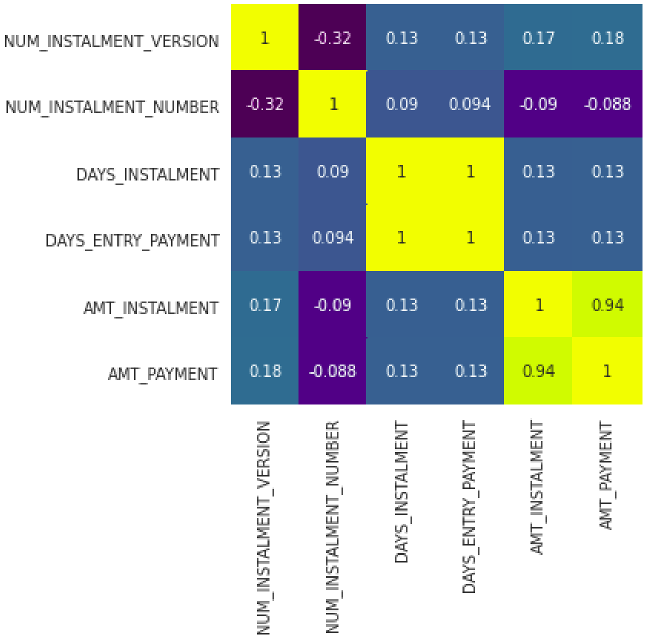

Fig. 7. Correlation Matrix for previous Installments

As we can see in the graph, When the instalment of previous credit was supposed to be paid is 100% correlated with When was the instalments of previous credit paid actually, what was the prescribed instalments amount of previous credit on this instalments is highly correlated with What the client actually paid on previous credit on this instalments.

## V. Training

### A. *Part A*

For Training, after feature engineering on application_train dataset, total of 130 original, encoded and extracted features were used. Since the dataset was unbalanced, Stratified Kfold is used for cross— validation with 5 and 10 folds.

Parameters Tuned for LightGBM are:

1. N-Estimators
2. Num_leaves
3. Max_depth
4. Learning rate
5. Boosting type
6. Min_data_in_leaf
7. Feature_fraction
8. Bagging_fraction
9. Bagging_frequency
10. Scale_pos_weight
11. Reg_alpha
12. Reg_lambda

**Analysis of Parameters:**

1. Speed: Change in bagging_fraction, bagging_freq, feature_fraction was resulting in increase in speed of training and hence consuming less time. With feature_fraction as 0.8 it randomnly selected 80% of the feature on each epoch, Thereby Making the model don't rely on specific set of features
2. Accuracy and Regularization: To Make sure model does not overfit, reg_alpha (L2 regularization parameter) is used as 0.5 and reg_lambda (L1 regularization parameter) is also used as 0.5. Since the tree grows level wise in lgbm, efforts are made to keep tree as sparse as possible by setting the max_depth at 16, which is much greater than num_leaves which is 511.
3. Three different boosting techniques (gbdt, goss and dart) are used.

The result on different parameters tuning are as follows:

TABLE I. VALIDATIOn ROC score on Part A

| *Model* | *Area Under ROC* |
|---|---|
| Random Forest | 0.70592 |
| LGBM :Dart | 0.74939 |
| LGBM: Goss | 0.75276 |
| LGBM: Stratified KFold | 0.75294 |

| *Model* | *Area Under ROC* |
|---|---|
| LGBM: Blend | 0.75692 |

## B. Part B: With Complete Dataset

Part B involved training the model on complete 7 dataset by joining each table with application_train and application_test on Primary Key Attribute “SK_ID_CURR”. But doing this requires a lot of memory as pandas by default assigns the largest possible datatype value. Therefore, it required to Modify the datatype of column depending upon the minimum and maximum value of column. The datatype of column where integer columns were modified to int8, int16, int32, int 64 while float columns were modified to float16, float32 and float 64. This saw a significant reduction in usage of memory

TABLE II. DATAFRAME SIZE VARIATION

| *Dataset* | *DataFrame Usage Before Modification* | *DataFrame Usage After Modification* |
|---|---|---|
| **Application_train** | 186 MB | 45MB |
| **Bureau_balance** | 624 MB | 624 MB |
| **Bureau** | 222 MB | 184 MB |
| **Credit_card_balance** | 673 MB | 275 MB |
| **Installments_payment** | 830 MB | 371 MB |
| **POS_CASH_balance** | 610 MB | 231 MB |
| **Prev_application** | 471 MB | 236MB |
| **TOTAL** | 3,616 MB | 1544 MB |

Since, there were multiple entries of “SK_ID_CURR” in remaining 6 datasets, “groupby” was performed and mean aggregate was taken. After Joining on Primary Key attribute “SK_ID_CURR”, similar steps were performed as mentioned in Part A, and result were as follows:

TABLE III. ROC SCORE FOR PART B

| *Model* | *Area Under ROC* |
|---|---|
| LGBM : GBDT | 0.77918 |
| LGBM: DART | 0.78425 |
| LGBM : GBDT with 30000 estimators | 0.78503 |

## C. Part C: Feature Engineering Most Important Features of Part B

**Step 1.** Feature engineering was done to create new features out of the most important features obtained after training the model on Part B.

Some of them are :

1. Income_credit_perc = Income / Credit
2. Family_Income= Income/COUNT_FAMILY_MEMBERs
3. Annuity_perc = Annuity / Income
4. Family Annuity = Annuity / Income / Count_of_Family_Members
5. Annuity Period = Credit / Annuity

**Step 2:** Instead of Taking just the mean of the other datasets while applying “groupby” on other 6 datasets, Min, Max, Median, Total and Variance was chosen.

**Step 3:** The 3 Model were trained on different parameters and Blending was done with Weighted Average Smoothing Technique.

TABLE IV. ROC SCORE FOR PART C

| Model | W1 | W2 | W3 | Area Under ROC |
|---|---|---|---|---|
| Vanilla | | | | 0.78809 |
| | 0.3 | 0.4 | 0.3 | 0.78881 |
| | 0.2 | 0.7 | 0.1 | 0.78884 |
| | 0.25 | 0.6 | 0.15 | 0.78890 |

The Fold ROC of Part C is shown in table below Table V and Importance of top 10 features is shown in fig 8.

TABLE V. TRAINING AND VALIDATION ROC SCORE FOR 5 FOLDS

| FOLD | Train_ROC | Validation_ROC |
|---|---|---|
| **1** | 0.9117 | 0.7852 |
| **2** | 0.8892 | 0.7817 |
| **3** | 0.9062 | 0.7806 |
| **4** | 0.9185 | 0.7830 |
| **5** | 0.8943 | 0.7796 |
| **Overall** | 0.9040 | 0.7819 |

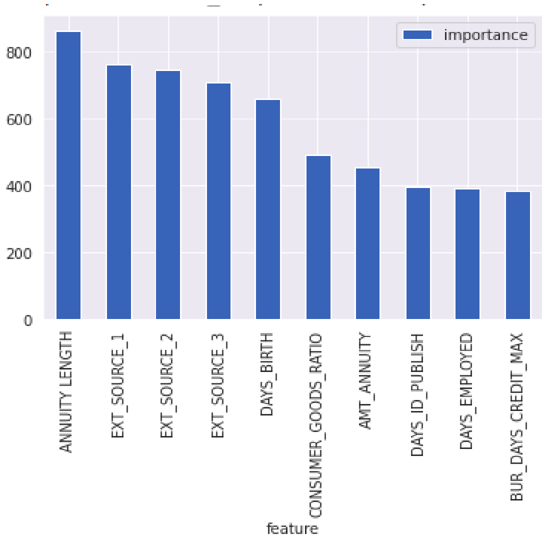

Fig. 8. Importance of Top 10 Features.

## VI. CONCUSION

Most of the time the Physical Memory is not enough to hold all the datasets in one go using Pandas DataFrame. This is due to the fact the pandas allocate the int64 or float 64 to all the attributes, even though the complete feature can be stored with int8 datatype. Therefore Manual modification of datatype is needed in order to ensure that complete DataFrame is loaded inside the memory. The best score is achieved with LightGBM with boosting technique as GBDT. Although change in hyperparameters is an essential part of training process, it did not saw significant increase in ROC score. Feature Engineering is an important step which did provide significant increase in Training as well as Test ROC score. The Average Weighted Technique is also an important tool as the training process is stochastic in nature and we cannot solely depend on one phase of training. It’s beneficial to take multiple instances of model and apply weighted average techniques.